

CLAIMS:

1 1. A host-fabric adapter, comprising:
2 at least one Micro-Engine (ME) arranged to establish connections and support data
3 transfers, via a switched fabric, in response to work requests from a host system for data
4 transfers;
5 interface blocks arranged to interface said switched fabric and said host system, and
6 send/receive work requests and/or data for data transfers, via said switched fabric, and configured
7 to provide context information needed for said Micro-Engine (ME) to process said work requests
8 for data transfers, via said switched fabric,
9 wherein said Micro-Engine (ME) is implemented with a pipelined instruction execution
10 architecture to handle one or more ME instructions and/or one or more tasks so as to process data
11 for data transfers.

1 2. The host-fabric adapter as claimed in claim 1, wherein said Micro-Engine (ME)
2 processes multiple ME instructions in parallel, when said ME instructions are deterministic logic
3 and arithmetic instructions by:
4 processing Instruction #1 at a first cycle in which an OpCode, source address and
5 destination address are read from an Instruction Memory;
6 providing a source address to the interface blocks for Instruction #1 at a second cycle, and
7 processing Instruction #2 in which the OpCode, source address and destination address are read
8 from the Instruction Memory;

1 when data for Instruction #1 is available from the interface blocks at a third cycle,
2 providing the source address to the interface blocks for Instruction #2, and processing Instruction
3 #3 in which the OpCode, source address and destination address are read from the Instruction
4 Memory;

5 processing data messages from the interface blocks for Instruction #1 at a fourth cycle,
6 and when data for Instruction #2 is available from the interface blocks, providing the source
7 address to the interface blocks for Instruction #3 and processing Instruction #4 in which the
8 OpCode, source address and destination address are read from the Instruction Memory;

9 providing destination and write controls of Instruction #1 for the interface blocks at a fifth
10 cycle, processing data messages from the interface blocks for Instruction #2, and when data for
11 Instruction #3 is available from the interface blocks, providing the source address to the interface
12 blocks for Instruction #4 and processing Instruction #5 in which the OpCode, source address and
13 destination address are read from the Instruction Memory; and

14 when Instruction #1 is retired at a sixth cycle, providing destination and write controls of
15 Instruction #2 for the interface blocks, processing the data from the interface blocks for
16 Instruction #3, and when data for Instruction #4 is available from the interface blocks, providing
17 the source address to the interface blocks for the instruction #5 and processing Instruction #6 in
18 which the OpCode, source address and destination address are read from the Instruction Memory.

1 3. The host-fabric adapter as claimed in claim 2, wherein said Micro-Engine (ME) is
2 configured to ensure that only latest data from the interface blocks is used and correct data is
3 written to the interface blocks.

1 4. The host-fabric adapter as claimed in claim 1, wherein said Micro-Engine (ME)
2 processes multiple ME instructions in parallel, when said ME instructions are non-deterministic
3 logic and arithmetic instructions by:

4 processing Instruction #1 at a first cycle in which an OpCode, source address and
5 destination address are read from an Instruction Memory;

6 providing the source address to the interface blocks for Instruction #1 at a second cycle,
7 and processing Instruction #2 in which the OpCode, source address and destination address are
8 read from the Instruction Memory;

9 when data for Instruction #1 is available from the interface blocks at a third cycle, and a
10 conditional Jump instruction based on Flags is set for Instruction #1, processing Instruction #3 in
11 which the OpCode, source address and destination address are read from the Instruction Memory;

12 processing data from the interface blocks for Instruction #1 at a fourth cycle, providing
13 the source address to the interface blocks for Instruction #3, processing Instruction #4 in which
14 the OpCode, source address and destination address are read from the Instruction Memory;

15 when data for Instruction #3 is available from the interface blocks at a fifth cycle,
16 providing destination and write controls of Instruction #4 for the interface blocks;

17 if the Jump condition is not TRUE, processing Instruction #5 in which the OpCode,
18 source address and destination address are read from the Instruction Memory;

19 if the Jump condition is TRUE, processing the conditional Jump instruction in which the
20 OpCode, source address and destination address are read from the Instruction Memory
21 corresponding to a Jump Address;

1 when Instruction #1 is retired at a sixth cycle, flushing Instruction #3 and data for
2 Instruction #4 available from the interface blocks, and providing the source address to the
3 interface blocks for the conditional Jump instruction corresponding to the Jump Address if the
4 Jump condition is TRUE; and

5 if the Jump condition is FALSE, providing the source address to the interface blocks for
6 Instruction #5 and processing the conditional Jump instruction in which the OpCode, source
7 address and destination address are read from the Instruction Memory corresponding to the Jump
Address.

5. The host-fabric adapter as claimed in claim 4, wherein said Micro-Engine (ME) is
configured to ensure that only latest data from the interface blocks is used and correct data is
written to the interface blocks.

6. The host-fabric adapter as claimed in claim 1, wherein said Micro-Engine (ME)
processes multiple tasks in parallel by:

3 processing Instruction #1 at a first cycle in which an OpCode, source address and
4 destination address are read from an Instruction Memory;

5 providing the source address to the interface blocks for Instruction #1 at a second cycle,
6 and processing Instruction #2 indicating a Task Switching Instruction in which the OpCode,
7 source address and destination address are read from the Instruction Memory;

1 when data for Instruction #1 is available from the interface blocks and there is no data
2 processing at a third cycle, processing Instruction #3 for a new task in which the OpCode, source
3 address and destination address are read from the Instruction Memory;

4 processing data for Instruction #1 from the interface blocks at a fourth cycle and
5 providing the source address to the interface blocks for Instruction #3 for the new task;

6 providing destination and write controls of Instruction #1 for the interface blocks at a fifth
7 cycle and, when data for the new task for Instruction #3 is available from the interface blocks,
8 providing the source address to the interface blocks for Instruction #4 and processing Instruction
9 #5 for the new task in which the OpCode, source address and destination address are read from
10 the Instruction Memory;

11 when Instruction #1 is retired at a sixth cycle, processing data from the interface blocks
12 for Instruction #3 for the new task, and when data for the new task for Instruction #4 is available
13 from the interface blocks, providing the source address to the interface blocks for Instruction #5
14 and processing Instruction #6 for the new task in which the OpCode, source address and
15 destination address are read from the Instruction Memory; and

16 when Instruction #2 is retired at a seventh cycle, providing destination and write controls
17 for the interface blocks for Instruction #3, processing data from the interface blocks for
18 Instruction #4 for the new task, and when data for the new task for Instruction #5 is available
19 from the interface blocks, providing the source address to the interface blocks for Instruction #6
20 and processing Instruction #7 for the new task in which the OpCode, source address and
21 destination address are read from the Instruction Memory.

1 7. The host-fabric adapter as claimed in claim 6, wherein said Micro-Engine (ME) is
2 configured to ensure that only latest data from the interface blocks is used and correct data is
3 written to the interface blocks.

1 8. The host-fabric adapter as claimed in claim 1, wherein said interface blocks
2 comprises:

3 a serial interface arranged to receive and transmit data from said switched fabric for data
transfers;

 a host interface arranged to receive and transmit work requests, in the form of work queue
elements (WQEs), from said host system for data transfers;

 a context memory arranged to store context information needed for said Micro-Engine
(ME) to process work requests for data transfers;

 a first-in/first-out (FIFO) interface arranged to receive data from said switched fabric via
said serial interface, and to transmit data to said switched fabric via said serial interface;

11 an address translation interface arranged for address translation from said Micro-Engine
12 (ME);

13 a local bus interface arranged to support system accessible context connections and data
14 transfers; and

15 a completion queue/doorbell manager interface arranged to provide an interface to
16 completion queues, and to update the context information needed for said Micro-Engine (ME) to
17 process work requests for data transfers.

1 9. The host-fabric adapter as claimed in claim 1, wherein said Micro-Engine (ME)
2 comprises:
3 one or more Data Multiplexers arranged to supply appropriate interface data based on an
4 ME instruction;
5 an Instruction Memory arranged to provide said ME instruction based on downloadable
6 MicroCode;
7 an Arithmetic Logic Unit (ALU) arranged to perform mathematical, logical and shifting
8 operations, and supply write data to the host interface, the address translation interface, the
9 context memory interface, the local bus interface, the completion queue/doorbell manager
10 interface, and the FIFO interface, via a system data bus; and
11 an Instruction Decoder arranged to supply system controls to the host interface, the
12 address translation interface, the context memory interface, the local bus interface, the
13 completion queue/doorbell manager interface, and the FIFO interface, via a system control bus,
14 to execute said ME instruction from said Instruction Memory to control operations of said Data
15 Multiplexers, and to determine functions of said Arithmetic Logic Unit (ALU).

1 10. The host-fabric adapter as claimed in claim 9, wherein said Instruction Memory
2 corresponds to a random-access-memory (RAM) provided to store MicroCode that are
3 downloadable for providing said ME instruction to said Instruction Decoder.

1 11. The host-fabric adapter as claimed in claim 10, wherein said Micro-Engine (ME)
2 and said interface blocks are configured in accordance with the "*InfiniBand™ Specification*", are
3 implemented as part of an Application Specific Integrated Circuit (ASIC).

1 12. A host-fabric adapter installed at a host system for connecting to a switched fabric
2 of a data network, comprising:

3 at least one Micro-Engine (ME) arranged to establish connections and support data
4 transfers via said switched fabric;

5 a serial interface arranged to receive and transmit data from said switched fabric for data
6 transfers;

7 a host interface arranged to receive and transmit work requests from said host system for
8 data transfers; and

9 a context memory interface arranged to store context information needed for said Micro-
10 Engine (ME) to process work requests for data transfers,

11 wherein said Micro-Engine (ME) is implemented with a pipelined instruction execution
12 architecture to handle one or more ME instructions and/or one or more tasks in parallel so as to
13 process data for data transfers.

1 13. The host-fabric adapter as claimed in claim 12, wherein said Micro-Engine (ME)
2 processes multiple ME instructions in parallel, when said ME instructions are deterministic logic
3 and arithmetic instructions by:

1 processing Instruction #1 at a first cycle in which an OpCode, source address and
2 destination address are read from an Instruction Memory;

3 providing a source address to the interface blocks for Instruction #1 at a second cycle, and
4 processing Instruction #2 in which the OpCode, source address and destination address are read
5 from the Instruction Memory;

6 when data for Instruction #1 is available from the interface blocks at a third cycle,
7 providing the source address to the interface blocks for Instruction #2, and processing Instruction
8 #3 in which the OpCode, source address and destination address are read from the Instruction
9 Memory;

10 processing data messages from the interface blocks for Instruction #1 at a fourth cycle,
11 and when data for Instruction #2 is available from the interface blocks, providing the source
12 address to the interface blocks for Instruction #3 and processing Instruction #4 in which the
13 OpCode, source address and destination address are read from the Instruction Memory;

14 providing destination and write controls of Instruction #1 for the interface blocks at a fifth
15 cycle, processing data messages from the interface blocks for Instruction #2, and when data for
16 Instruction #3 is available from the interface blocks, providing the source address to the interface
17 blocks for Instruction #4 and processing Instruction #5 in which the OpCode, source address and
18 destination address are read from the Instruction Memory; and

19 when Instruction #1 is retired at a sixth cycle, providing destination and write controls of
20 Instruction #2 for the interface blocks, processing the data from the interface blocks for
21 Instruction #3, and when data for Instruction #4 is available from the interface blocks, providing

1 the source address to the interface blocks for the instruction #5 and processing Instruction #6 in
2 which the OpCode, source address and destination address are read from the Instruction Memory.

1 14. The host-fabric adapter as claimed in claim 13, wherein said Micro-Engine (ME)
2 is configured to ensure that only latest data from the interface blocks is used and correct data is
3 written to the interface blocks.

1 15. The host-fabric adapter as claimed in claim 12, wherein said Micro-Engine (ME)
2 processes multiple ME instructions in parallel, when said ME instructions are non-deterministic
3 logic and arithmetic instructions by:

4 processing Instruction #1 at a first cycle in which an OpCode, source address and
5 destination address are read from an Instruction Memory;

6 providing the source address to the interface blocks for Instruction #1 at a second cycle,
7 and processing Instruction #2 in which the OpCode, source address and destination address are
8 read from the Instruction Memory;

9 when data for Instruction #1 is available from the interface blocks at a third cycle, and a
10 conditional Jump instruction based on Flags is set for Instruction #1, processing Instruction #3 in
11 which the OpCode, source address and destination address are read from the Instruction Memory;

12 processing data from the interface blocks for Instruction #1 at a fourth cycle, providing
13 the source address to the interface blocks for Instruction #3, processing Instruction #4 in which
14 the OpCode, source address and destination address are read from the Instruction Memory;

1 when data for Instruction #3 is available from the interface blocks at a fifth cycle,
2 providing destination and write controls of Instruction #4 for the interface blocks;
3 if the Jump condition is not TRUE, processing Instruction #5 in which the OpCode,
4 source address and destination address are read from the Instruction Memory;
5 if the Jump condition is TRUE, processing the conditional Jump instruction in which the
6 OpCode, source address and destination address are read from the Instruction Memory
7 corresponding to a Jump Address;

8 when Instruction #1 is retired at a sixth cycle, flushing Instruction #3 and data for
9 Instruction #4 available from the interface blocks, and providing the source address to the
10 interface blocks for the conditional Jump instruction corresponding to the Jump Address if the
11 Jump condition is TRUE; and

12 if the Jump condition is FALSE, providing the source address to the interface blocks for
13 Instruction #5 and processing the conditional Jump instruction in which the OpCode, source
14 address and destination address are read from the Instruction Memory corresponding to the Jump
15 Address.

1 16. The host-fabric adapter as claimed in claim 15, wherein said Micro-Engine (ME)
2 is configured to ensure that only latest data from the interface blocks is used and correct data is
3 written to the interface blocks.

1 17. The host-fabric adapter as claimed in claim 12, wherein said Micro-Engine (ME)
2 processes multiple tasks in parallel by:

1 processing Instruction #1 at a first cycle in which an OpCode, source address and
2 destination address are read from an Instruction Memory;

3 providing the source address to the interface blocks for Instruction #1 at a second cycle,
4 and processing Instruction #2 indicating a Task Switching Instruction in which the OpCode,
5 source address and destination address are read from the Instruction Memory;

6 when data for Instruction #1 is available from the interface blocks and there is no data
7 processing at a third cycle, processing Instruction #3 for a new task in which the OpCode, source
8 address and destination address are read from the Instruction Memory;

9 processing data for Instruction #1 from the interface blocks at a fourth cycle and
10 providing the source address to the interface blocks for Instruction #3 for the new task;

11 providing destination and write controls of Instruction #1 for the interface blocks at a fifth
12 cycle and, when data for the new task for Instruction #3 is available from the interface blocks,
13 providing the source address to the interface blocks for Instruction #4 and processing Instruction
14 #5 for the new task in which the OpCode, source address and destination address are read from
15 the Instruction Memory;

16 when Instruction #1 is retired at a sixth cycle, processing data from the interface blocks
17 for Instruction #3 for the new task, and when data for the new task for Instruction #4 is available
18 from the interface blocks, providing the source address to the interface blocks for Instruction #5
19 and processing Instruction #6 for the new task in which the OpCode, source address and
20 destination address are read from the Instruction Memory; and

21 when Instruction #2 is retired at a seventh cycle, providing destination and write controls
22 for the interface blocks for Instruction #3, processing data from the interface blocks for

1 Instruction #4 for the new task, and when data for the new task for Instruction #5 is available
2 from the interface blocks, providing the source address to the interface blocks for Instruction #6
3 and processing Instruction #7 for the new task in which the OpCode, source address and
4 destination address are read from the Instruction Memory.

1 18. The host-fabric adapter as claimed in claim 17, wherein said Micro-Engine (ME)
2 is configured to ensure that only latest data from the interface blocks is used and correct data is
written to the interface blocks.

19. The host-fabric adapter as claimed in claim 12, wherein said Micro-Engine (ME)
comprises:

one or more Data Multiplexers arranged to supply appropriate interface data based on an
ME instruction;

an Instruction Memory arranged to provide said ME instruction based on downloadable
MicroCode;

an Arithmetic Logic Unit (ALU) arranged to perform mathematical, logical and shifting
operations, and supply write data to the host interface, the address translation interface, the
context memory interface, the local bus interface, the completion queue/doorbell manager
interface, and the FIFO interface, via a system data bus; and

an Instruction Decoder arranged to supply system controls to the host interface, the
address translation interface, the context memory interface, the local bus interface, the
completion queue/doorbell manager interface, and the FIFO interface, via a system control bus,

1 to execute said ME instruction from said Instruction Memory to control operations of said Data
2 Multiplexers, and to determine functions of said Arithmetic Logic Unit (ALU).

1 20. The host-fabric adapter as claimed in claim 19, wherein said Micro-Engine (ME)
2 and said appropriate interface are configured in accordance with the "*InfiniBand™ Specification*",
3 are implemented as part of an Application Specific Integrated Circuit (ASIC).